




Serdar Zuli Software Engineer

Date of birth: 25/09/1997

CONTACT

 serdar.zuli@gmail.com

 (+389) 72531700

 <https://thezuli.com>

 www.linkedin.com/in/serdar-zuli-98335a183/

ABOUT ME

I am a Computer Engineering graduate from Çanakkale University with over 3 years of experience in software development. I have worked on various projects in web development, artificial intelligence, and cloud solutions. I am passionate about learning new technologies and finding innovative solutions.

EDUCATION AND TRAINING

2017 – 2022 Çanakkale, Türkiye

Bachelor's Degree Çanakkale Onsekiz MART University

Website <http://ce.muhendislik.comu.edu.tr/>

WORK EXPERIENCE

03/2024 – 09/2024

Software developer Werner & Kollegen

Project: Automated Accident Assessment System

Developed an innovative automated accident assessment system designed to streamline the post-accident process for drivers. The system enables users to determine liability in a car accident quickly and efficiently without the need for police reports or court proceedings.

AI and Cloud Projects:

- **Model Development with Google Vertex AI:** Developed machine learning models using TensorFlow and PyTorch on Google Vertex AI.
- **Text Processing with Google Document AI:** Performed text processing tasks using Google Document AI.

Backend Development:

- **.NET Core-based Project Structure:** Played a key role in designing and developing the backend of the system using .NET Core. Established a robust project structure adhering to industry best practices.
- **SOLID Principles:** Applied SOLID principles to ensure maintainable and scalable code.
- **Clean Architecture Pattern:** Implemented the Clean Architecture pattern to separate concerns and create a flexible and testable codebase.
- **Domain-Driven Design:** Defined domain entities, created repositories, and developed the business layer to manage application logic effectively.
- **Database Design:** Structured and implemented the database, establishing relationships between tables to ensure data integrity and performance.

End-to-End Involvement: Involved in the complete development lifecycle, from initial design and architecture to implementation and deployment.

Microservices Architecture:

- Integrated models and text processing services through microservices to ensure efficient communication between different components.
- Created endpoints to provide seamless access to end-users.

Technical Responsibilities:

- Used Python for machine learning model development on Google Vertex AI.
- Utilized C# for implementing text processing with Google Document AI.
- Developed and managed APIs to allow users to interact with AI services effectively.

Website: <https://west.cloud>

12/2023 – 03/2024 London

Software engineer CDRL

In this project, I developed a solution to process and categorize a large dataset of 6000 images using advanced AI techniques. The project involved several key steps:

- **Azure OCR for Data Extraction:** Used Azure OCR to accurately read and extract textual information from the images.
- **Azure AI for Image Classification:** After extracting the text, I classified the images into relevant categories using Azure AI, optimizing the process for efficiency and accuracy.
- **GPT-4 Integration:** Used GPT-4 to automatically write the extracted data from the images into an Excel file for easier analysis.
- **AI Model Development:** Utilized TensorFlow to train a machine learning model with the categorized data for further processing and classification.

This project showcased my ability to integrate cloud AI services with machine learning techniques, delivering a robust and scalable solution for large-scale image classification tasks.

for more details visit: https://thezuli.com/ocr_azure_ai_tools & https://thezuli.com/vertex_ai_details

Cloud Platforms:

- Azure Vision AI
- Document Service AI,
- OCR
- Blob Container
- GPT4 API

Python:

- AI & Machine Learning: TensorFlow, PyTorch
- Web Technologies: Flask
- MicroServices
- OOP Principles
- API Development
- Queue Management: RabbitMQ
- Databases: MongoDB

Website: <https://www.cdrl.org.uk>

06/2023 – 12/2023 Las Vegas, United States

Software developer Pictor

In this project, I developed a robust face swap application using Python, built from the ground up with a microservices architecture. The system is designed to efficiently handle requests via RabbitMQ for queue management. I implemented Docker containers and app services on Azure DevOps to ensure seamless integration and deployment across all components. The project includes a RESTful API (FLASK) endpoint, which I thoroughly tested under high-load conditions using Apache JMeter to ensure performance and reliability.

For more details visit: https://thezuli.com/faceswap_details

Roles and Responsibilities:

- Developed a face-swapping application using Python and microservices architecture.
- RabbitMQ: Managed high-volume requests with RabbitMQ for queue management.
- Docker and Azure: Used Docker containers and Azure DevOps for smooth deployment.
- Load Testing: Tested the system under heavy load with Apache JMeter.

Technologies and Tools:

- Python
- Flask env
- Asyncio
- Garbage Memory
- Open-source Image Processing Libraries
- Azure Blob Container
- Azure Web Service
- Azure Docker Container
- RabbitMQ
- Database: Azure Cloud
- Microservice structure: (Python & ASP.NET CORE)

Website: <https://app.pictor.pro/onboarding>

09/2022 – 05/2023 Amsterdam, Netherlands

Software developer Imperum

Roles and Responsibilities:

Developed and implemented automated triggers to ensure the security of incoming data and verify the reliability of accessed websites.

Utilized technologies such as Cisco, Wazuh, and VirusTotal endpoints to check if files were infected with viruses.

Designed and managed fully automated background triggers to enhance security measures.

Achievements:

Successfully ensured the security and reliability of data and websites through advanced trigger mechanisms.

Integrated multiple security technologies to provide comprehensive virus detection and prevention.

Improved system security and user trust by implementing robust verification processes.

Technologies and Tools:

- Developed automated triggers for security checks and website reliability.
- Technologies: Integrated Cisco, Wazuh, and VirusTotal for advanced virus detection.
- Python & Flask: Built RESTful APIs for security systems using Python and Flask.

Python:

- Asyncio lib
- Strawberry lib
- RESTful API
- Flask Env
- GraphQL
- Mongo DB

Connectors:

- Cisco
- Wazuh
- VirusTotal
- CrowdStrike
- GateWatcher

Website: <https://imperum.io>

SKILLS

Technical Skills

- Microservice's,
- Onion Architecture,
- Solid Pattern,
- Mediator pattern,
- OOP Principles,

- SignalR,
- Hubs,
- Queue Management (RabbitMQ).
- Web Frameworks: ASP.NET, Flask.
- API Integration: Designing and integrating RESTful and SOAP web services.

- Asynchronous I/O
- Asyncio
- Databases: PostgreSQL, MongoDB, SQL Server.

- Artificial Intelligence: TensorFlow, PyTorch, VertexAI
- Cloud Computing: Experience with cloud platforms like AWS, Azure, and Google Cloud.
- DevOps: Skills in CI/CD tools (Azure DevOps), and containerization (Docker).

- Software Development Methodologies: Experience with Agile, and Scrum methodologies.

Programming Languages

- C# Advanced (ASP.NET CORE)
- Python Advanced

Database

- SQL (Microsoft SQL Server, PostgreSQL)
- NoSQL (MongoDB, Redis)

Operating systems

- Linux
- Windows
- Mac

Business and Communication Skills

- Problem Solving: Ability to identify and resolve technical issues.
- Teamwork: Effective collaboration and working within a team.
- Communication Skills: Ability to clearly explain technical information to non-technical stakeholders.
- Project Management: Experience in planning, managing, and delivering projects
- Attention to Detail: Precision and accuracy in detail-oriented tasks.

LANGUAGE SKILLS

MOTHER TONGUE(S): Turkish

Other language(s):

English

Listening C1

Spoken production B2

Reading C1

Spoken interaction C1

Writing B2

Macedonian

Listening C1

Spoken production C1

Reading C1

Spoken interaction C1

Writing C1

Albanian

Listening C1

Spoken production C1

Reading C1

Spoken interaction C1

Writing B1

Levels: A1 and A2: Basic user; B1 and B2: Independent user; C1 and C2: Proficient user